Rational

the e-development company

UML para Factorías

Capítulo 8: Relaciones entre Clases y Objetos

Objetivos: Relaciones

- **Al final de este capitulo, usted podrá:**
  - Identificar y representar en diagramas de clases, los tipos mas importantes de relaciones entre clases:
    - Asociación
    - Agregación
    - Generalización
  - Usar nombres de asociación y roles para hacer mas claras las asociaciones
  - Definir y especificar la multiplicidad y navegabilidad de una asociación
  - Definir y representar una asociación o agregación reflexiva y clases de asociación

La Necesidad de las Relaciones

- **En el “dominio del problema”, muchas veces existen conexiones semánticas entre dos o mas objetos, que deben representarse como parte del Modelo de Análisis**
  - La representación de una conexión semántica se logra usando relaciones
- **Por otro lado, los objetos contribuyen a definir el comportamiento de un sistema, al colaborar unos con otros**
  - La colaboración se puede lograr a través de relaciones
- **Existen tres tipos de relaciones que se pueden considerar para representar en el análisis:**
  - Asociación
  - Agregación
  - Generalización



Asociación

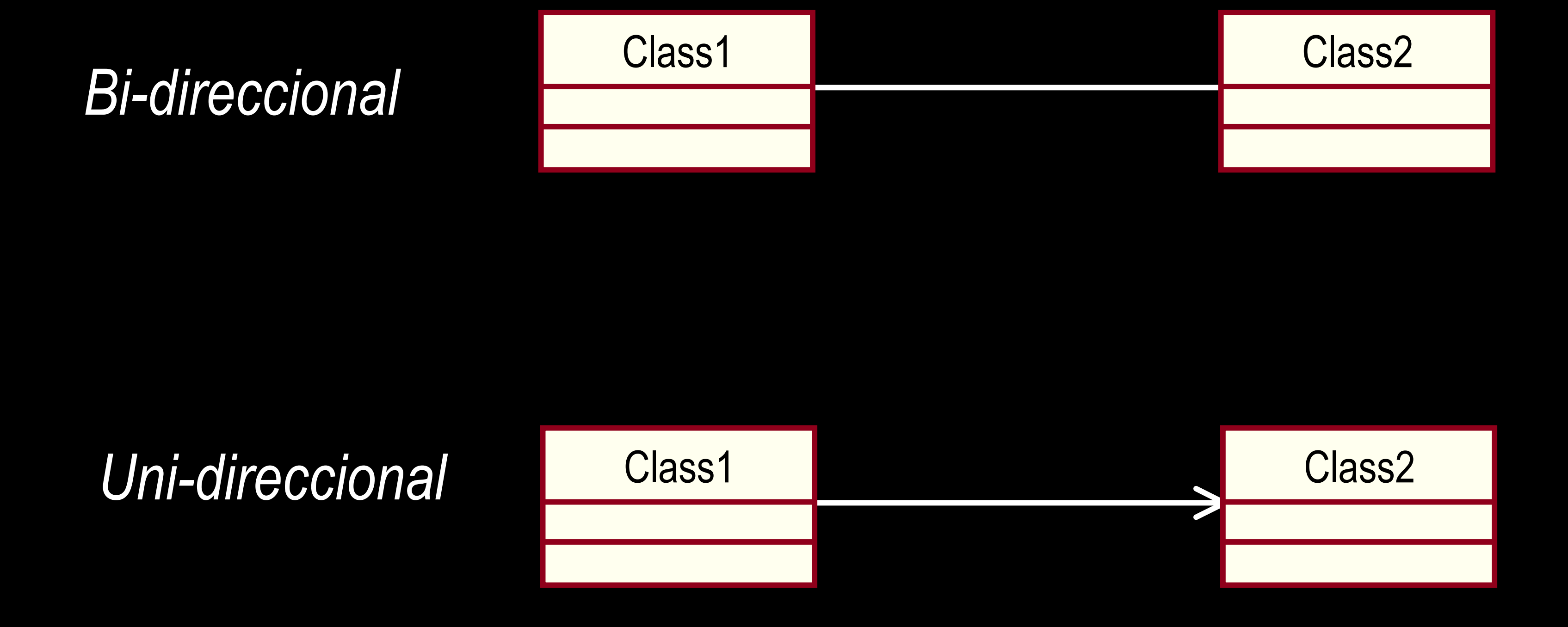
- **La asociación es la forma mas común de representar una conexión semántica entre dos clases**
  - Implica que existe un enlace entre los objetos de las clases asociadas
- **En el Análisis, se usa la asociación para relacionar dos objetos que necesitan colaborar enviándose mensajes**
  - Los mensajes (y la información asociada a estos) pueden navegar en una u otra dirección a través de una asociación
- **Las asociaciones se representan en los diagramas de clases con una línea que conecta las clases asociadas**





Navegabilidad

- **Indica la posibilidad de navegar de una clase asociada a una clase objetivo**



Navegabilidad

- **Una asociación es por defecto una relación bidireccional**
  - Dada una instancia de Sección existe un objeto :Curso asociado
  - Dada una instancia de Curso existe un objeto :Sección asociado



Nombrando Asociaciones

- **Para aclarar su significado, se le puede dar un nombre a una asociación**
- **El nombre se representa con una etiqueta colocada sobre la línea de asociación en medio de los íconos de las clases**
- **Usualmente un nombre de asociación es un verbo o una frase con verbos**



Definiendo Roles

- Un rol denota el propósito o capacidad en la que una clase se asocia con otra
- Típicamente los nombres de los roles son sustantivos o frases sustantivas
- El nombre de un rol se coloca sobre la línea de asociación cerca la clase a la que modifica
- Uno o ambos extremos de la asociación pueden tener nombres

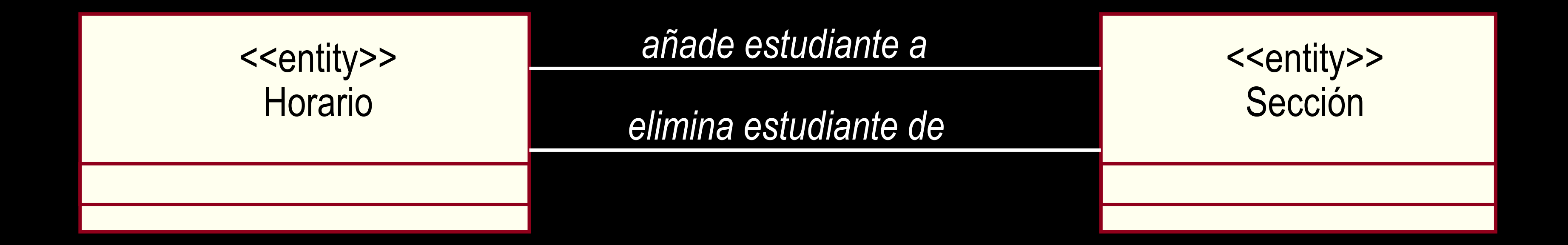


Asociaciones Múltiples

- **Esta situación se da cuando entre dos clases existe más de una asociación**
- **Las asociaciones múltiples deben reflejar múltiples roles**
  - Los objetos de estas clases deben tener una conexión semántica por cada asociación
  - No debe definirse una asociación distinta para cada mensaje que se pase entre dos clases que colaboran
- **Si hay más de una asociación entre dos clases entonces es OBLIGATORIO nombrarlas**



Asociaciones Múltiples



¿Es esto correcto?



Asociaciones múltiples *(cont.)*

<<entity>>
Horario

añade estudiante a

elimina estudiante de

<<entity>>
Sección

<<entity>>
Horario

Cursos Primarios

Cursos Alternos

<<entity>>
Sección

Las asociaciones múltiples deben reflejar múltiples roles



Multiplicidad para Asociaciones

- **La Multiplicidad es el número de instancias de una clase relacionadas a UNA instancia de otra clase**
- **Para cada asociación existen dos decisiones que tomar en cuanto a multiplicidad; una para cada extremo de la asociación**
- **Por ejemplo, en la conexión entre Profesor y Curso**
  - Cada instancia de Profesor, puede impartir muchos (cero o más) cursos
  - Para cada instancia de Curso, hay exactamente un Profesor



Indicadores de Multiplicidad

- **Cada extremo de la asociación contiene un indicador de multiplicidad**
  - Indica el número de objetos que participan en la relación

Sin especificar	
Exactamente uno	1
Muchos (Cero o más, sin limite)	*
	0..*
Uno o más	1..*
Cero o uno	0..1
Rango Específico	2..4
Varios Rangos	2,4..6,8

¿Qué significa la Multiplicidad?

- **La multiplicidad responde dos preguntas**
  - ¿La asociación es obligatoria o es opcional?
  - ¿Cuál es el mínimo y máximo número de objetos de una clase que se pueden enlazar a un objeto de la otra clase?



- **Las decisiones de multiplicidad exponen muchas suposiciones escondidas acerca del problema que se está modelando**
  - ¿Se puede ir el maestro de licencia o tomar vacaciones?
  - ¿Puede un curso tener dos maestros?



Agregación

- **La agregación es una forma especializada de asociación en la que un "agregado" (el todo) está relacionado con su(s) parte(s)**
  - La agregación se conoce como una relación "parte-de" o relación de contención-contenido
- **Una agregación se representa como una asociación con un diamante vacío en el extremo de la clase que denota el agregado**
- **La multiplicidad se representa de la misma forma que otras asociaciones**

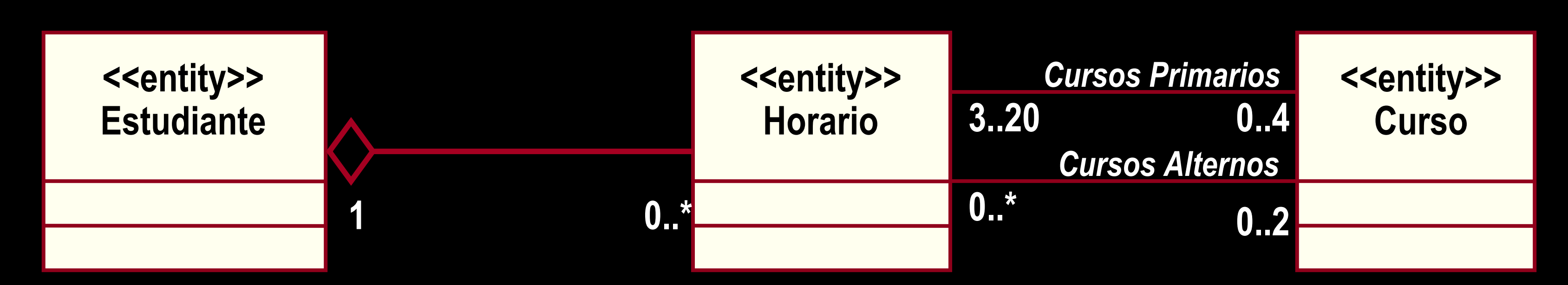


Pruebas de Agregación

- **¿La frase "parte de" es usada para describir la relación?**
  - Una Puerta es "parte de" un Carro
- **¿Algunas operaciones en el todo automáticamente se aplican a sus partes?**
  - Mover el Carro, Mover la Puerta
- **¿Se propagan algunos de los valores de los atributos del todo a todas o algunas de sus partes?**
  - El Carro es azul, la Puerta es azul
- **¿Existe una asimetría intrínseca en la relación donde una clase es subordinada a la otra?**
  - Una Puerta ES parte de un Carro, un Carro NO ES parte de una Puerta

¿Asociación o Agregación?

- **Si los dos objetos están firmemente unidos por una relación todo-parte**
  - La relación es una agregación
- **Si los dos objetos se consideran usualmente como independientes, aún cuando muchas veces se enlazan**
  - La relación es una asociación
- **Si hay duda**
  - Escoger asociación



Contenimiento

- **En una relación de agregación, el objeto origen (agregado) debe "contener" conocimiento semántico (have containment) del objeto destino (parte)**
- **Las relaciones de agregación pueden ser de dos tipos según como contengan los agregados a las partes:**
  - Contenimiento por Referencia: es la forma mas comun de agregacion y en ella el objeto agregado contiene solo identificadores con los que pueda hacer referencia a los objetos partes (por ejemplo una lista de punteros). Esto quiere decir que los objetos relacionados pueden "vivir" independientemente.
  - Contenimiento por Valor: a la agregacion de este tipo se les conoce también como composición, e implica que los objetos partes estan contenidos dentro de los objetos agregados y son parte de su estructura.

Composición

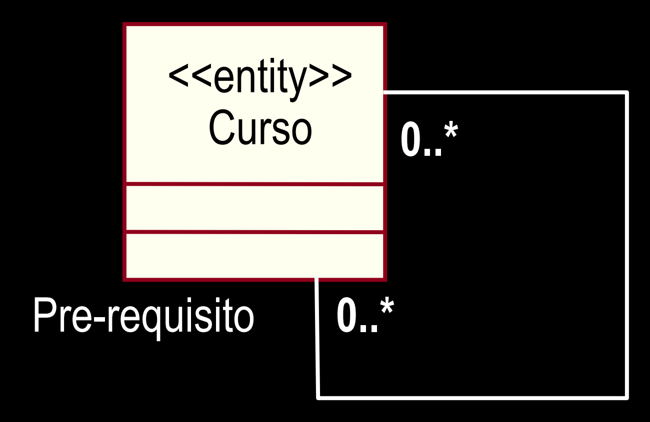
- La composicion representa el contenimiento estructural de los objetos parte dentro de los objetos agregados relacionados
- Implica que existe una dependencia entre los tiempos de vida de los objetos y posiblemente deban de crearse y/o destruirse conjuntamente
- Se representa como una asociación con un diamante lleno en el extremo de la clase que denota el agregado





Asociaciones Reflexivas

- **Se usa una asociación reflexiva para modelar una situación en donde existen enlaces entre objetos de la misma clase**
- **Esta relación básicamente indica que dos o mas objetos en la misma clase colaboran juntos de alguna manera**

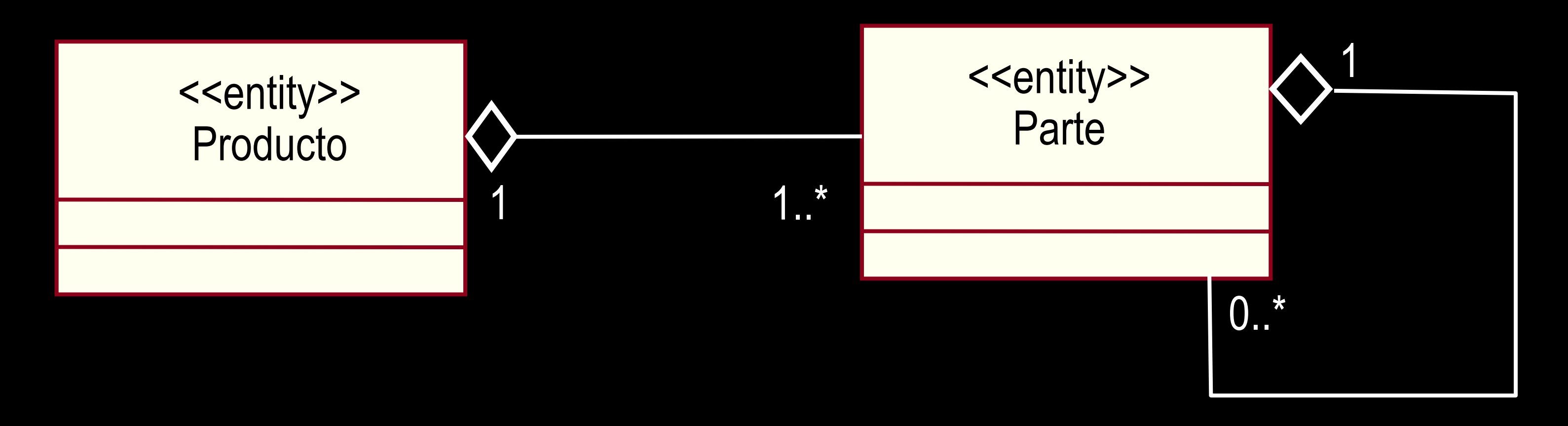


Agregados Reflexivos

- **Las asociaciones de agregación también pueden ser reflexivas**
  - Problema clásico de “explosión de materiales”

Un objeto :Producto se “compone de” uno o mas objetos :Parte

Un objeto :Parte se “compone de” cero o más objetos :Parte



Información en la Relación

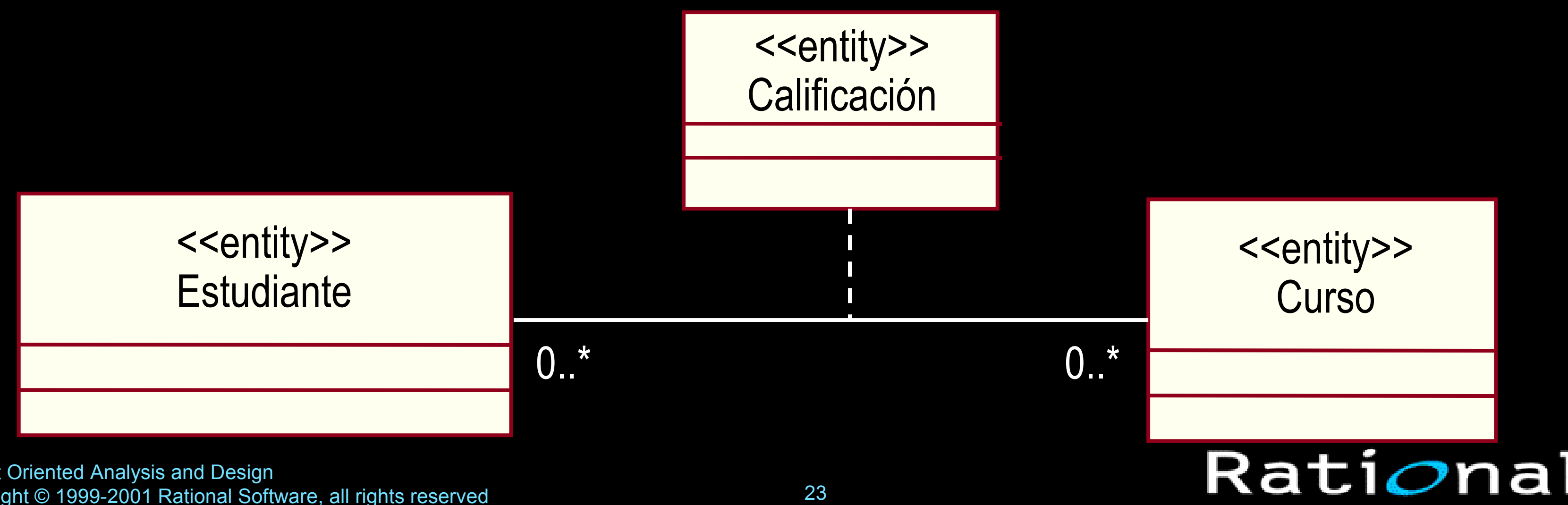
- **En algunas situaciones, se encuentra que la información que se quiere incluir en el modelo no pertenece a ningún objeto especifico, sino a una relación entre dos objetos**
- **Ejemplo:**
  - Deseamos llevar control de las calificaciones para todos los cursos que el estudiante ha tomado y por esto creamos una relación “muchos-a-muchos” entre las clases Estudiante y Curso ¿Dónde se coloca el atributo calificación?
  - El atributo calificación no puede colocarse en la clase Curso por que hay (potencialmente) muchos enlaces a muchos objetos Estudiante
  - El atributo calificación no se puede incluir en la clase Estudiante por que hay (potencialmente) muchos enlaces a muchos objetos Curso
  - Por lo tanto, el atributo pertenece realmente a enlace individual Estudiante-Curso





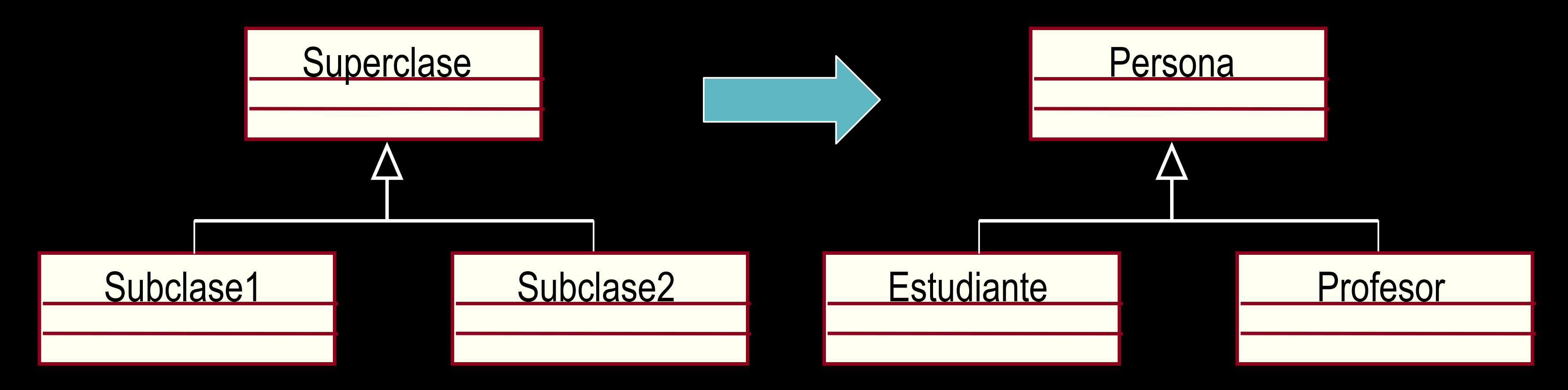
Clases de Asociación

- **Las clases de asociación se utilizan para modelar la información que pertenece a un enlace entre dos clases**
  - Una clase de asociación puede incluir múltiples propiedades (atributos)
  - Se permite solo una clase de asociación por asociación
  - Las clases de asociación se usan comúnmente para resolver relaciones de “muchos-a-muchos” como las tablas intermedias en un ERD
- **En UML, las clases de asociación se representan como clases que están conectadas a la asociación a que pertenecen con una línea intermitente**



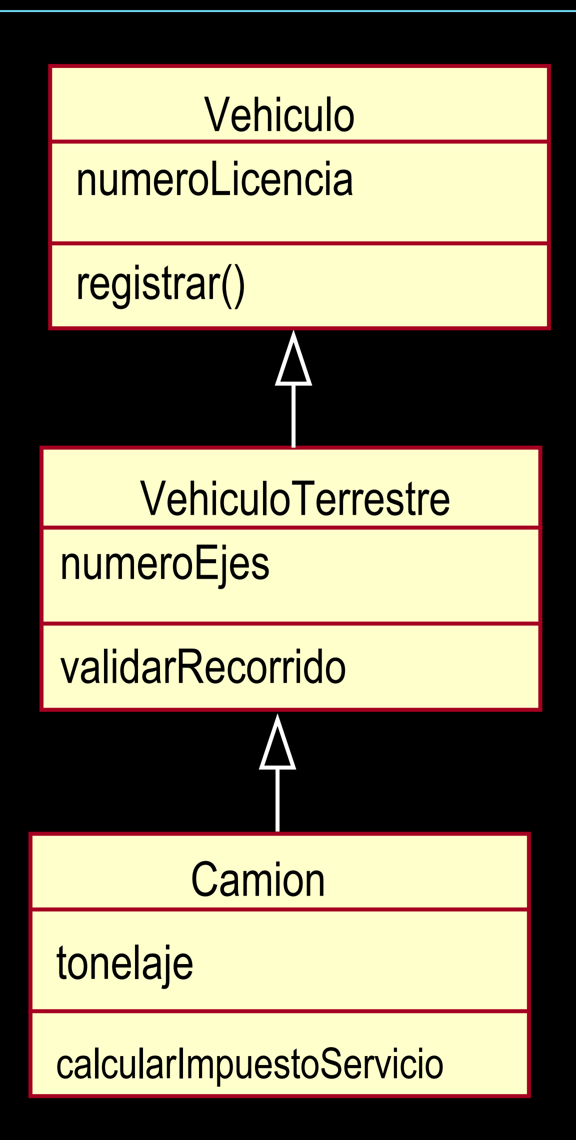
Generalización

- **Una generalización es una relación entre clases en donde una clase comparte su estructura y/o comportamiento con una o mas clases en niveles distintos de abstracción**
- **Las relaciones de generalización se basan en el concepto básico de OO que llamamos Jerarquía y su principal característica es el uso de la Herencia**
  - Permiten modelar relaciones entre “superclases” y “subclases”, o desde el punto de vista de la herencia, entre “clases padres” y “clases hijas”
  - Una subclase es “un tipo de” su superclase



¿Que se hereda en la Generalización?

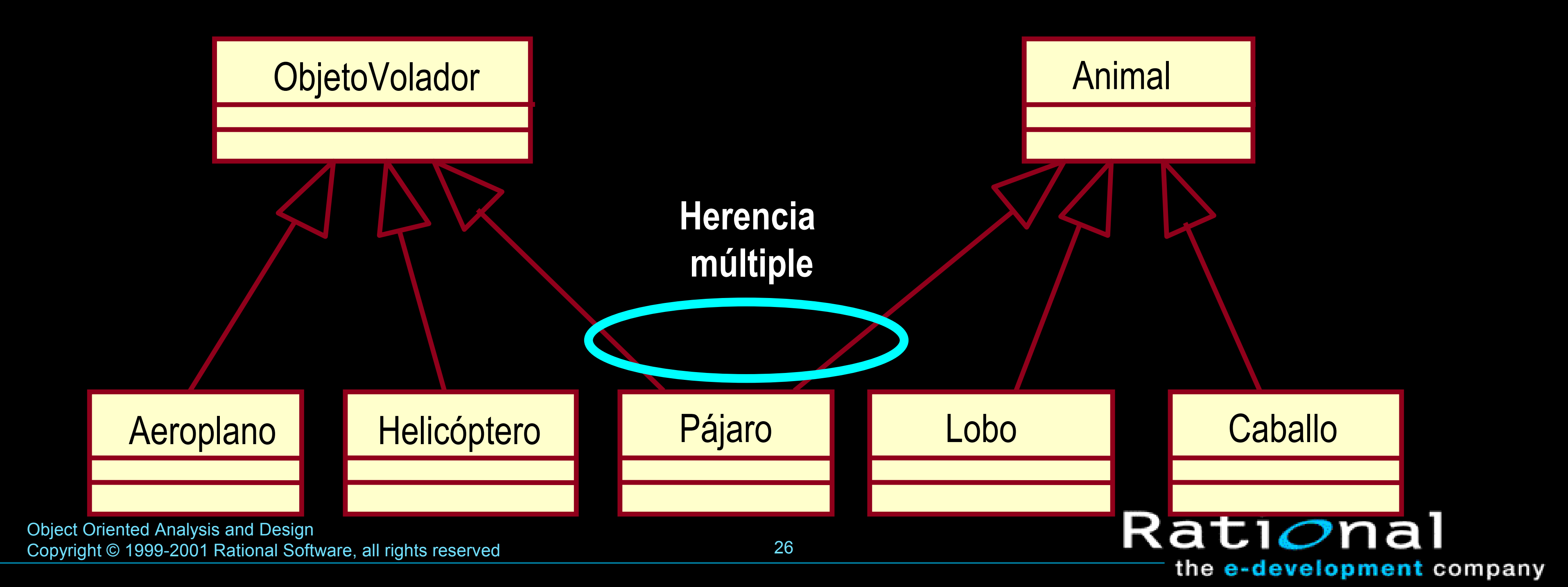
- Las subclases heredan las propiedades (atributos) y el comportamiento (operaciones) de las superclases
- Una jerarquía de generalización puede tener varios niveles y la herencia se mantiene en todos estos





Herencia Múltiple

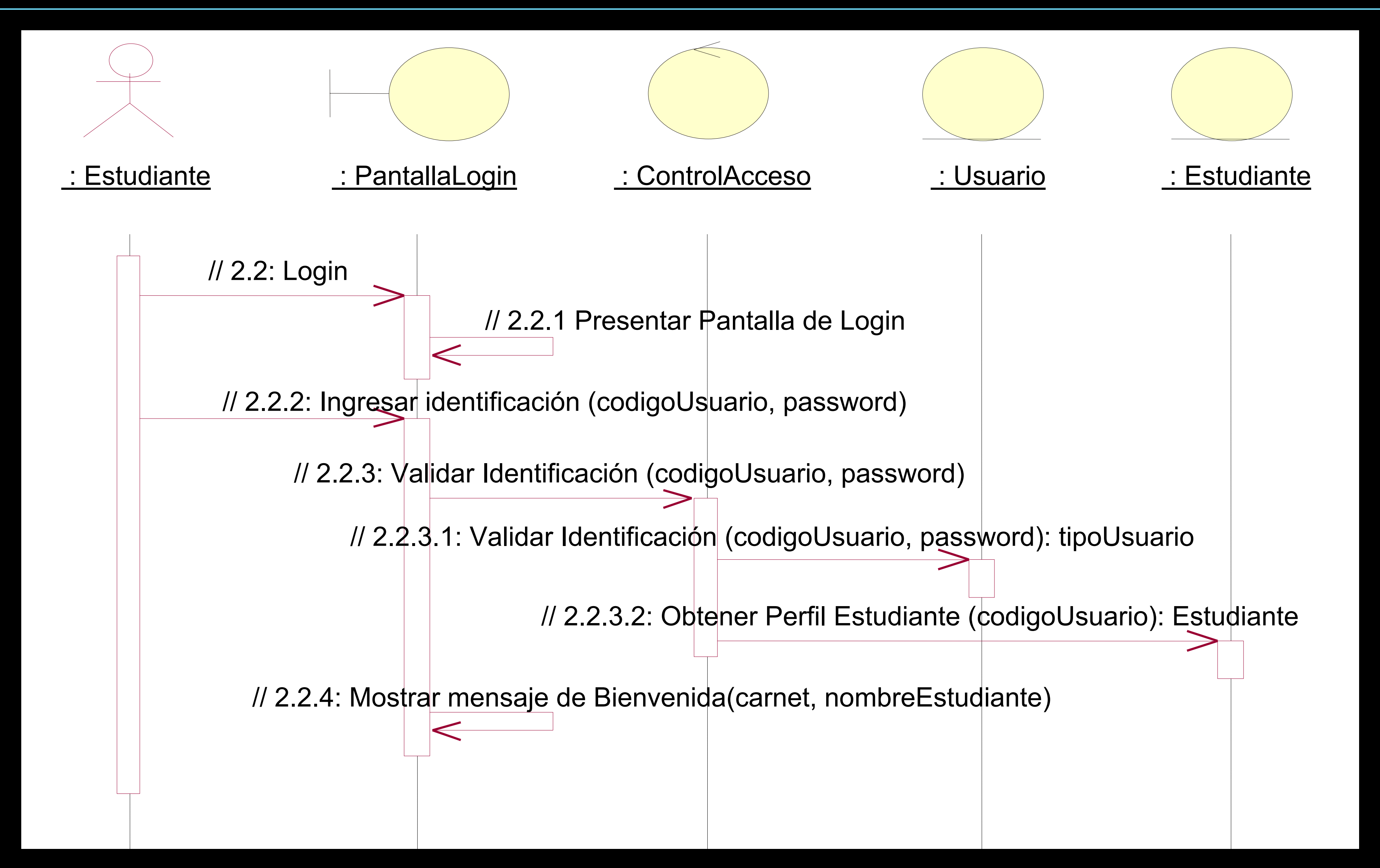
- La mayoria de las veces una clase hereda de solo otra clase (herencia simple), sin embargo una misma clase puede heredar de varias clases distintas (herencia multiple).
- Esto puede ayudar a simplificar el Modelo de Analisis, pero debe usarse juiciosamente, ya que puede ser dificil de implementar en el Modelo de Diseño.



Encontrando Relaciones por Colaboración

- La relaciones definidas para representar las conexiones semánticas entre las clases deben complementarse mediante el Análisis de Casos de Uso:
  - Al examinar los escenarios de casos de uso en los diagramas de interacción, podemos determinar que objetos deben colaborar entre si enviándose mensajes.
  - Esta colaboración puede darse solo si los objetos se "conocen" para poder comunicarse.
  - Puede usarse la relacion de asociacion para establecer esta ruta de comunicación; al relacionar las clases de los objetos que colaboran entre si, se asegura que los escenarios se pueden completar.
  - Esto significa que podemos complementar la definición de las relaciones para el Modelo de Analisis, con una asociacion por cada grupo de mensajes que exista entre los objetos de las clases que se estan asociando.

Ejemplo - CU “Login” – Escenario “Estudiante”



Ejemplo - CU “Login” – Escenario “Estudiante”

Encontrando Relaciones por Colaboración:

- Del mensaje
  “// 2.2.3: Validar Identificación (codigoUsuario, password)”
  sabemos que la clase PantallaLogin puede asociarse a la clase ControlAcceso

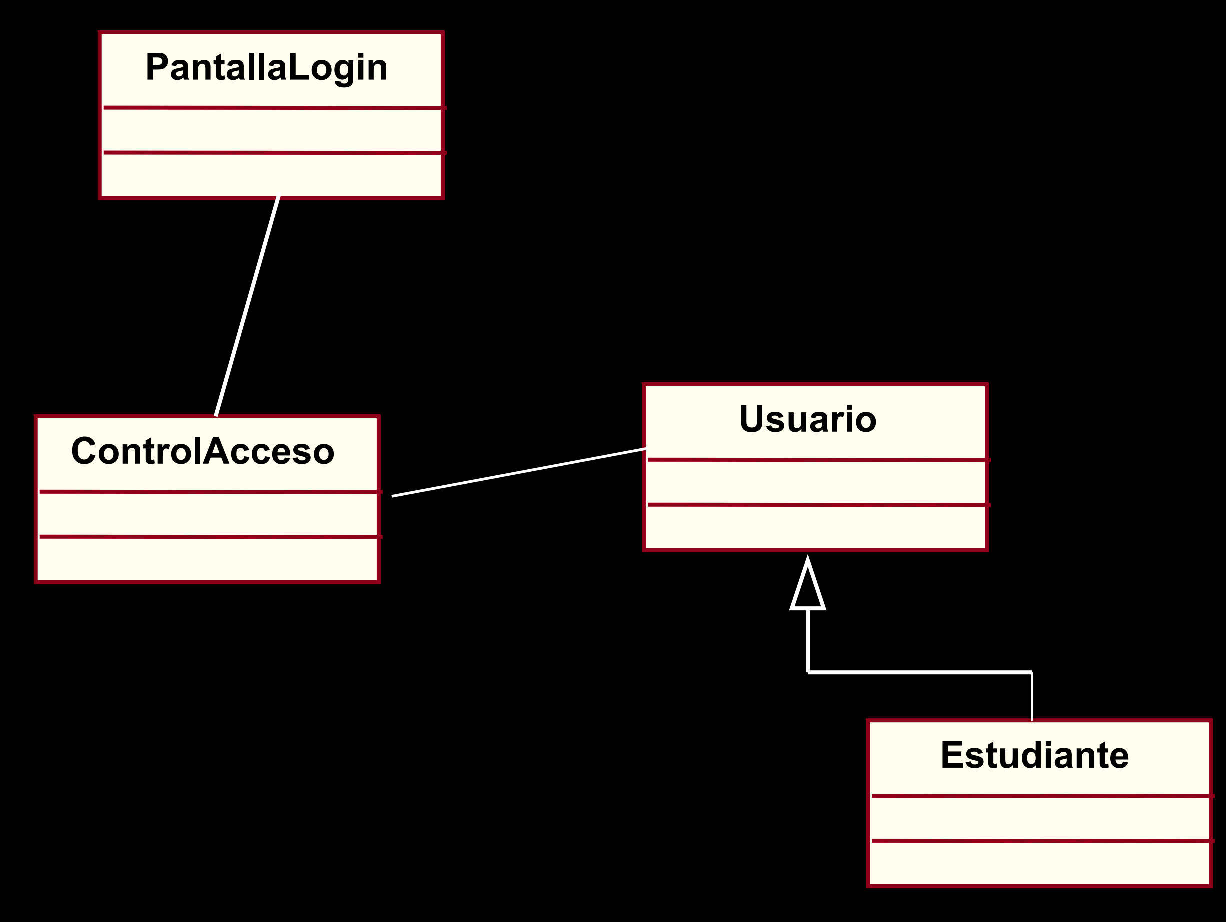
- De los mensajes
  “// 2.2.3.1: Validar Identificación (codigoUsuario, password): tipoUsuario”
  “// 2.2.3.2: Obtener Perfil Estudiante (codigoUsuario): Estudiante”
  sabemos que la clase ControlAcceso puede asociarse a las clases Estudiante y Usuario

- Además podemos considerar que existe una conexión semántica entre las clases Usuario y Estudiante, que puede representarse como una generalización. Esto significa que no es necesario asociar ControlAcceso a Estudiante, ya que esta implícito en la relación entre Estudiante y Usuario.

Ejemplo - CU “Login” – Escenario “Estudiante”

Encontrando Relaciones por Colaboración:

- En un diagrama de clases, esto lo representamos:



Descubriendo Relaciones Adicionales en las Operaciones|

- **Los argumentos de las operaciones y las clases de retorno pueden denotar una relación entre la clase que contiene la operación, y la clase argumento y/o la clase de retorno**
- **Ejemplo:**
  - La clase ListadoCurso tiene una operación agregarEstudiante(Juan:InfoEstudiante)
  - Esto implica que existe una relación entre ListadoCurso e InfoEstudiante

Relaciones en el Modelo de Análisis y el Modelo de Diseño

- **Para el Modelo de Análisis:**
  - Se deben establecer las conexiones semánticas entre clases para representarlas como asociaciones, agregaciones y generalizaciones. Estas conexiones existen por la naturaleza de las clases, no por una implementación específica.
  - Se pueden definir relaciones de asociación entre las clases cuyos objetos colaboran comunicándose entre si.
  - Debe hacerse un estimado inicial de multiplicidad para poner en evidencia suposiciones escondidas.
- **Para el Modelo de Diseño**
  - Los estimados de multiplicidad se afinan y actualizan.
  - Las asociaciones, agregaciones y generalizaciones se evalúan y actualizan.
  - Se maduran los diagramas de clase.



Resumen: Relaciones

- Los objetos de las distintas clases de un sistema pueden relacionarse entre si porque existe una conexión semántica entre ellos, o porque deben colaborar dentro del contexto de un escenario de casos de uso.
- Los 3 tipos de relaciones que deben representarse en el Modelo de Análisis son son las asociaciones, agregaciones y generalizaciones.
- Una asociación es una conexión entre dos clases que representa comunicación
  - Una asociación puede tener nombre
  - Se pueden usar nombres de roles
  - La comunicación puede ser uni- o bi-direccional (por defecto)

Resumen: Relaciones *(cont.)*

- La multiplicidad es el número de instancias que participan en una asociación
  - Es representada al final de una línea de asociación
  - Cada extremo de una asociación puede tener un indicador de multiplicidad
- Una agregación es una forma especializada de asociación en la que el todo está relacionado con sus partes
  - Cada extremo de la línea de agregación debe tener un indicador de multiplicidad
- Una clase puede tener una asociación reflexiva
  - Se da cuando dos objetos de la misma clase que están relacionados
- Las relaciones de agregación también pueden ser reflexivas
  - Problemas de tipo lista de materiales (partes que están formadas de partes)

Resumen: Relaciones *(cont.)*

- La información que pertenece al enlace entre objetos se muestra como una clase de asociación
- La generalización permite definir jerarquías de herencia entre clases que están a distintos niveles de abstracción.
- La colaboración entre los objetos de dos clases puede tomarse como criterio para definir una relación de asociación entre las dos clases.
- Las relaciones que se representan en el Modelo de Análisis son una versión inicial de las que pueden incluirse en el Modelo de Diseño.